3D MODEL CONTROL OF IMAGE PROCESSING

An H. Nguyen and Lawrence Stark

Telerobotics Unit

University of California at Berkeley

INTRODUCTION

Telerobotics studies remote control of distant robots by a human operator using supervisory or direct control. Even if the robotic manipulator has vision or other senses, problems arise involving control, communications, and delay [18]. The communication delays that may be expected with telerobots working in space stations while being controlled from an Earth laboratory have led to a number of experiments attempting to circumvent the problem (Fig. 1). This delay in communication is a main motivating factor in moving from well-understood instantaneous hands-on manual control to less well-understood supervisory control [5,7]; the ultimate step would be the realization of a fully autonomous robot.

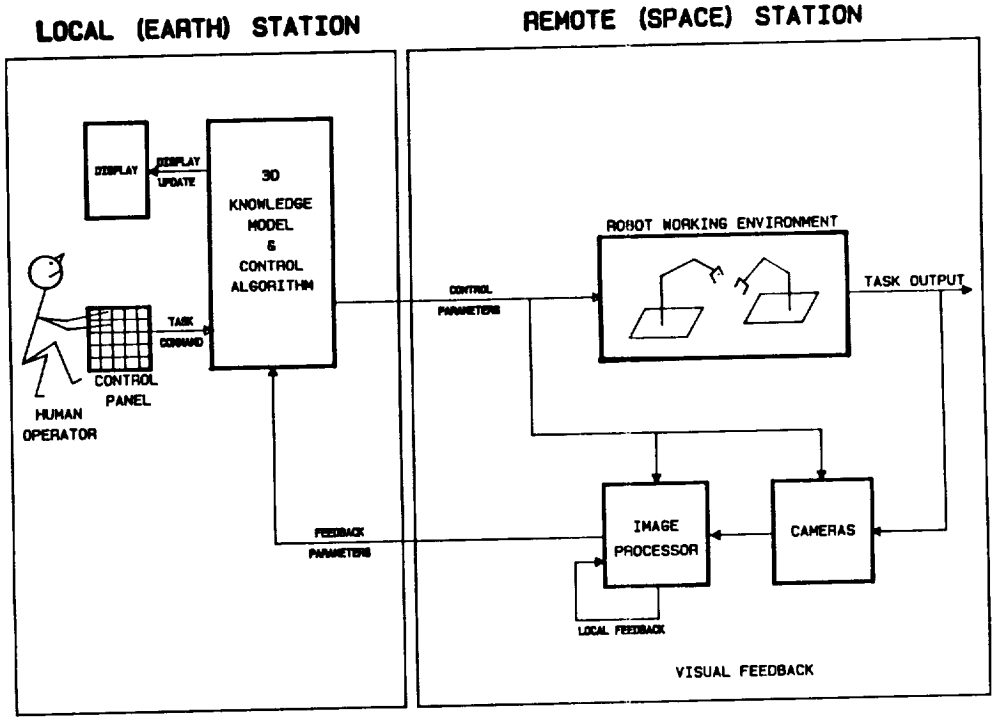

Fig. 1. Overview sketch of model control of robot working environment and of image processing

METHODS

Hardware Setup: Two a-robots (Armatron robots), modified to interface with an AT-386 computer [13,20] via parallel I/O ports, are controlled by computer in an autonomous mode. Manual control capability is preserved for teaching and for the supervisory mode, since hands-on control is vital in developing and evaluating different control algorithms. Both a-robots operate within a one-cubic-meter working environment. One of the robots (painted dark blue) is fitted with ONSNEs features (Fig. 2, left panel). Three cameras were used; one, an inexpensive C-mounted TV camera (Panasonic, model WV-1410); the others, commercially available 8mm camcorders (Sanyo, model VM-10). They provided two orthogonal side and top views, and an oblique view (Fig. 2, left panel). The computer selected among the camera views by means of a four-channel video multiplexer, whose output was connected to a simple frame grabber (Epix-Silicon Video, Chicago). The frame grabber resided in the AT bus and was directly controlled by the computer to digitize video images into 320 x 240 arrays of 8-bit pixels.

Software: Besides the main program performing administrative work, three major pieces of software were developed to control the mobile a-robot in obtaining a given target with visual feedback. These consisted of the ICM, 3DM, and utility programs. The ICM program included many different low-level image processing algorithms such as edge enhancement, feature extraction, automatic thresholding, filtering, moments computation. The 3DM program supported a complete, scaled-down model of the a-robot and its RWE. It also provided 2D projections of different camera views, and contained an algorithm for simple path planning. The utility software was highly optimized, and consisted of all the primitive functions for the frame grabber, EGA graphic display and plotter. All software was written in "C".

RESULTS: THE 3D MODEL

At the local earth station, the human operator views a display of the 3D model and uses the control panel in a supervisory mode to oversee the control algorithms (Fig. 1). At the remote space station, the control parameters drive the robots in the robot working environment (RWE). These control parameters also drive the cameras and the image processing algorithms. Besides a local feedback process, the main feedback is from the remote image processing to the Earth station 3D model.

A remote RWE is modeled using graphics workstation (Iris) with 3D graphic transformation support hardware (Fig. 3). At the RWE, three a-robots perform tasks; the m-robot (Mitsubishi manipulator) holds a camera and actively searches for optimum views. This experimental set-up provides us with a global view of the telerobotics control situation wherein several robots cooperate in a joint task, or each robot has an individual task assigned (Fig. 3). The 3D model is constructed from information about the robotic manipulators, the work pieces, and the camera positions [17]. The 3D model guides the image processor in extracting information derived from regions-of-interest (ROI) which contain on-the-scene visual enhancements (OSCNE) (Fig. 2, left); note the model with on-the-screen visual enhancements (OSCRN) [10,11] for use by the human operator (Fig. 2, right).

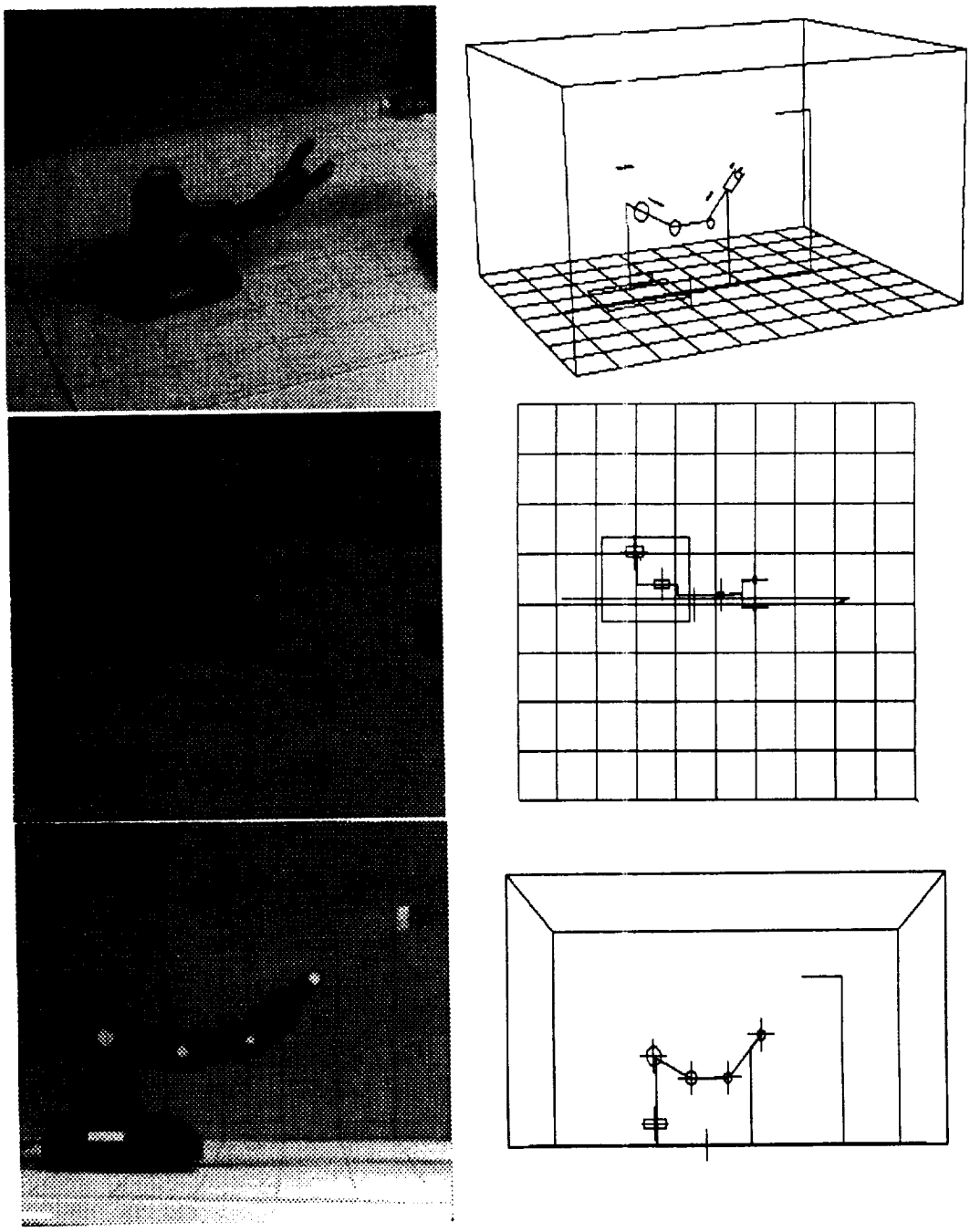

Fig. 2 Perspective views and orthogonal projections of a-robot (left) and model (right) showing on-the-scene visual enhancements (OSCNE). 3D model guides image processor to extract information only in regions-of-interest.

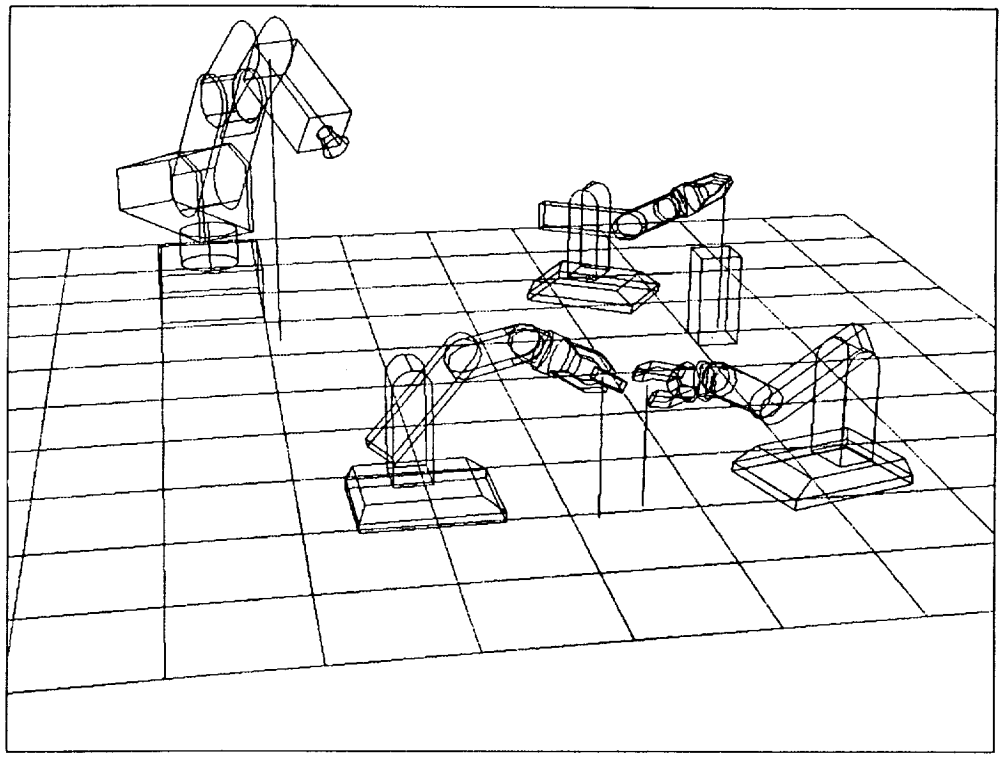

Fig. 3. Perspective view of vectorgraphic model of three a-robots and one m-robot.

Feature_selection_criteria_and_ROIs_locations: An important step in using top-down image processing to control robots is to specify the most useful information to be gained from the robot and its environment, and to determine the physical location of these features. These selections strongly influence not only the choice of information processing strategies, but also the image processing schemes employed.

Supposing that links of a robot are known or fixed, the kinematic recovery of the 3D robot model simply requires the joint location information. Any two consecutive joints of a robot provide complete information about the link length and orientation. In situations where the robot joint is not visible to the system, link orientation becomes important. In complex, multiple-robot environments, the image processing computer faces the far more challenging problems of occlusions, light reflections, shadows, noises, etc. Although the model uses a priori knowledge that plays an active role in resolving many of these problems, image processing tasks can be further simplified by introducing ONSNEs to both robots and the RWE (Fig. 4). These ONSNEs boost video signal to noise ratios within ROIs, and also may provide redundant information depending on their sizes and shapes.

Assignment_of_ROIs_locations: Each orthogonal projection view of the robot and its RWE has two sets of ROIs, the primary and secondary sets (Fig. 4). For the side view, the primary set (Fig. 4, upper) of ROIs is responsible for information about robot joints, while the secondary set of ROIs determines the robot orientations (Fig. 4, lower). Under static conditions, sizes of ROIs depend on those of ONSNEs. Since processing time is directly proportional to ROIs areas, ROIs should be small to minimize processing time, yet large enough to cover individual ONSNEs within ROIs. For automatic thresholding, optimum-size ROIs areas would be twice that of ONSNEs.

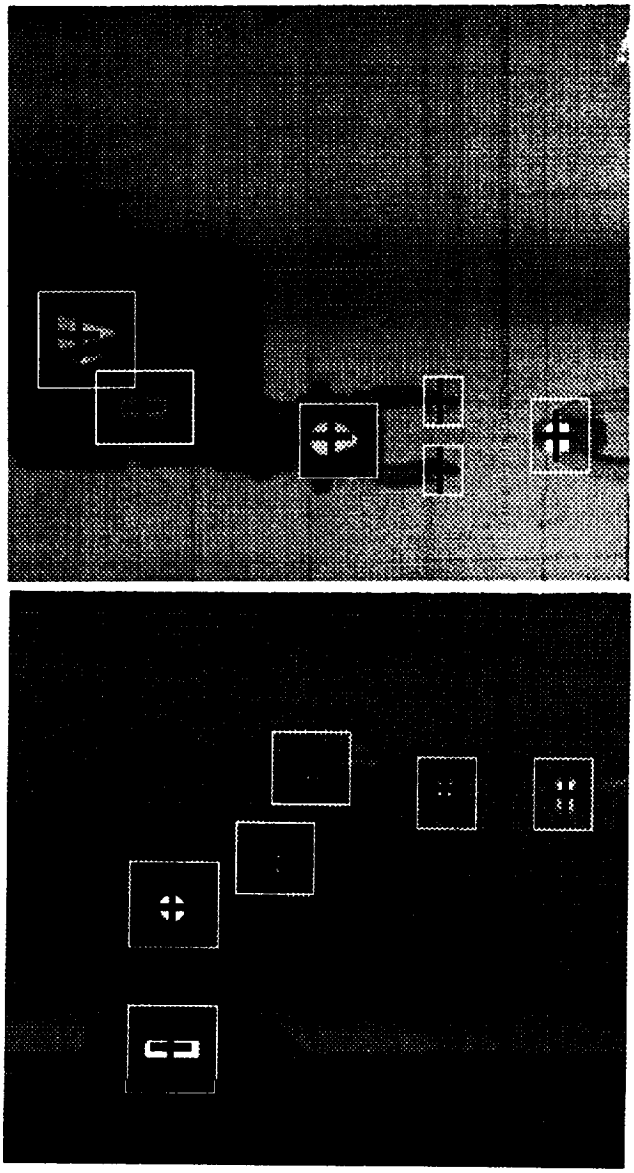

Fig. 4 Orthogonal views of actual robot reaching a target. While centroids of OSNCEs, resided within ROIs, provide feedback information for model to guide robot to target.

RESULTS : 2D IMAGE PROCESSING

Image processing operating within ROIs: As mentioned earlier, selected features determined both the locations of ROIs and the image processing schemes within them [13,19]. For instance, detecting angles of a robot link includes two steps: edge enhancement and line detection. The edge enhancement operation accentuates edges and acts like a two-dimensional high pass filter. Edge detection has been an active area of research for many years, and this continues today. Several algorithms for edge detection, such as Sobel, Kirsch, Roberts, and the Laplacian [6,8,14,16], are available and already implemented in VLSI devices [15]. These are simple operators in the form of 3 x 3 matrices. Another edge detection algorithm also worth mentioning is the Laplacian of the Gaussian [12]. This algorithm detects local edges effectively, and has been proven to be an optimum operator in dealing with true edges and noisy images [3]. Unfortunately, this operator requires a much larger kernel and, therefore, is computationally expensive. Since Sobel operators operate in pairs (the x and y directions independently), noise tends to be suppressed in one direction, while edges are accentuated in the other direction [15]. Due to their insensitivity to noises, simplicity in implementation, and efficiency in operation, the Sobel operators were incorporated into our scheme for low-level image processing in detecting edges.

Enhanced edges, resulting after Sobel operation, contain much higher intensity levels than the average. Therefore, appropriate threshold levels can be easily found, either by manual selection aided by histogram displays, or automatically by a thresholding algorithm. Threshold operations transform a gray level image into a binary image with two levels of intensity. Only enhanced edges above the threshold level remain after thresholding and are then ready for line detection. Orientation of a line can be retrieved by a number of algorithms such as matched filters, cross-correlation, or the Hough transform. Among these techniques, the Hough transform combined with top-down information from the model renders line information quite reliably and efficiently.

Centroid moment processing for the primary set of ROIs: Visual information residing in the primary set of ROIs provides sufficient feedback information for model adjustment and correction. Image processing carried out for this set of ROIs takes precedence over many other tasks, including control of the robots (Fig. 3, right panel; Fig. 4). Because of the strategic importance of this critical joint information, the ONSNEs were introduced (Fig. 3). The ONSNEs yield higher contrast in the video images, and thus more reliable visual information can be obtained under various luminance conditions.

Centroid and other invariant moments: The ONSNEs also have had a strong influence on the selection of the low-level image processing < 3 scheme used --- the invariant moments, a method in which centroids are derived. This technique had been previously applied to pattern recognition for printed characters [1,9], to chromosome analysis [2,4], and biological instrumentation [20]. The first three order moments yield information about size, centroid location, and major axis orientation for a bounded object; they are simple in implementation and inexpensive in computation. Additional higher order moments are also available for shape description, features that cannot be acquired from other low-level imaging schemes. Furthermore, the centroid parameters provide excellent information for local feedback (see Fig. 1), a special requirement for our image processing scheme.

Moments are widely utilized in classical mechanics; moments of a distribution function are also commonly used in statistical theory. For a given bounded, two-dimensional function f(x,y), the set of moments is defined as

$$M_{i,j} = \iint x^i y^j f(x,y)\, dx\, dy, \qquad i, j = 0, 1, 2, \ldots \tag{1}$$

In the infinte set [Mi,j] moments, as i and j take all non-negative values, uniquely determining the function f(x,y); and conversely, f(x,y) uniquely determines the set [Mi,j]. [i+j] is the order of the moment.

For binary images in which intensity of the object bounded by f(x,y) is one and zero elsewhere, the zeroth order moment M00

$$M_{00} = \iint f(x,y)\, dx\, dy \tag{2}$$

is the area of the object. Coordinates of the centroid are found to be,

$$\begin{aligned} xc &= M_{10}/M_{00} \\ yc &= M_{01}/M_{00} \end{aligned} \tag{3}$$

where M10 and M01 are the first moments for x and y respectively. Moments computed after translation of the origin to the center of gravity, are called central moments,

$$m_{ij} = \iint (x-xc)^i\, (y-yc)^j\, f(x,y)\, dx\, dy \tag{4}$$

For digitial image processing, equations (1) and (4) above, become

$$M_{ij} = \sum_x \sum_y x^i y^j\, f(x,y) \tag{5}$$

$$m_{ij} = \sum_x \sum_y (x-x_c)^i\, (y-y_c)^j\, f(x,y) \tag{6}$$

The second order central moments are [6]

$$m_{11} = M_{11} - y_c\, M_{10} \tag{7.a}$$

$$m_{20} = M_{20} - x_c\, M10 \tag{7.b}$$

$$m_{02} = M_{02} - y_c\, M01 \tag{7.c}$$

The object orientation or principal axis of rotation about this axis causes the second-order central moments to vanish [8,9]

$$\text{theta} = (\tan^{-1}\, (2m_{11}/(m_{20}-m_{02}))\, /\, 2 \tag{8}$$

All the area-normalized central moments relative to this principle axis are invariant under magnification, location, and rotation of the object [6,9].

RESULTS: AUTONOMOUS CONTROL

Control robot sequence: There are a number of different paths via which the robot can reach the target. However, for the purpose of this study, we derived a simple but effective algorithm to enable the 3D model to control the robot and to direct the image processing computer. The scheme worked satisfactorily regardless of initial positions and orientations of both robot and target. The process to reach a target consisted of two phases, the orientation phase and acquiring phase. To reach a designated target, the robot

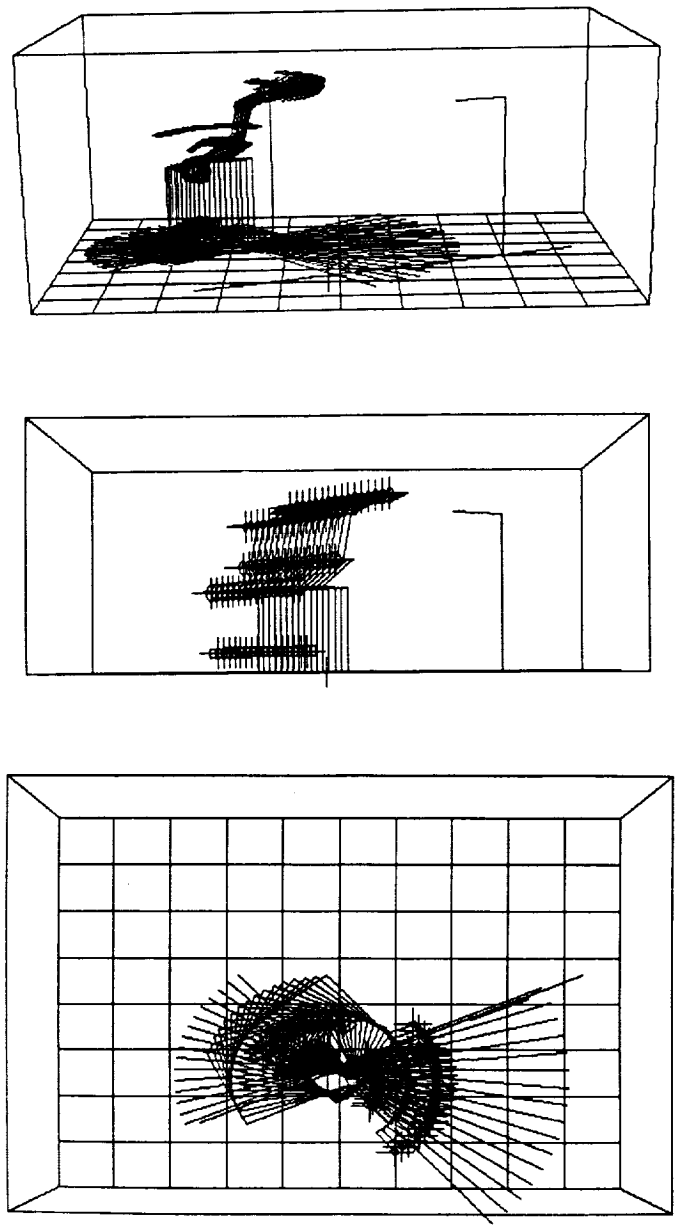

Fig. 5. Model approaching target: In the autonomous mode, upon receiving command to reach target, robot first performs orientation and then position relative to the target. Top: robot rotates to safe zone location. Middle: Then moves forward to this point. Bottom: Rotates to align with target.

first rotated (Fig. 5, upper view) until its new direction intersected with target direction at the safe zone location (Fig. 5; Fig. 3, right panel --- large plus signs superimposed on target approach line). Next the robot moved forward until it reached this location (Fig. 5, middle view). It then performed a second rotation so that its direction aligned with the target direction (Fig. 5, lower view). Thus the robot completed the first orientation phase in approaching the target. Since accuracy was not crucial in this phase, visual update was more relaxed and faster control speed could be obtained. Once the direction of the robot was in alignment with its target, the second phase began. The image processing computer immediately switched to a fast operating mode, closing the feedback loop. The 3D knowledge model carefully cruised the robot and guided the robot gripper to finally acquire the target (Fig. 2,4,5).

DISCUSSION

Since all tasks have been performed by the AT-386, our programs have grown close to the limit of the MS-DOS capability. Compromises have had to be made among competitive issues such as performance, memory utilization and implementation of new schemes. To alleviate this problem, the 3D knowledge-based model will be ported over to the SUN-386 workstation acting as the local control station (Fig. 1). It will oversee the image processing tasks and the control of the robots that will remain with the AT 386 computer in the remote station. The Iris graphics workstation will provide the display to the human operator at the earth laboratory.

Future research will include systematic benchmnark studies for the various image processing schemes as they fail while becoming subject to extreme conditions. Controllable cameras and wider working environments for the robots will also be utilized.

In conclusion, the top-down approach [20] with 3D model control plays a crucial role in resolving many conflicting image processing problems that are inherent in the bottom-up approach of most current machine vision processes. The 3D model control approach is also capable of providing the necessary visual feedback information for both the control algorithms and for the human operator. Finally, it provides an extreme reduction in communication, the mostly needed feature in telerobotics applications.

ACKNOWLEDGEMENTS: - We wish to thank Dr. A. K. Bejczy and Dr. Stephen Ellis, the technical monitors of our JPL and NASA/Ames research grants, and as well our colleagues in the Telerobotics Unit, UCB.

REFERENCES

1. Alt, F.L., "Digital Pattern Recognition by Moments", JACM 9:240-258 (1962).
2. Butler, J.W., Butler, M.K., and Stroud, A., "Automatic Classification of Chromosomes," K. Enslein, ed., Data Acquisition and Processing in Biology and Medicine, 3, Pergamon, New York (1964)
3. Canny, John, "A Computational Approach to Edge Detection", IEEE Trans. on Pattern Anal. and Machine Intel. PAMI-8 (1986).
4. Castleman, K. R., Digital Image Processing, Prentice Hall (1979).
5. Ferrell, W.R., Sheridan, T.B., "Supervisory Control of Remote Manipulation", IEEE Spectrum 4: 81-88 (1967).
6. Hall, E.L., Computer Image Processing & Recognition, Academic, N. Y. (1979).

7. Hirose, M., Sudo, J., and Ishii, T., "Human Interface of Remote Operation in Three-Dimensional Space," 2nd Symposium on Human Interface. Tokyo (1986).
8. Horn, B. K. P., "Robot Vision," MIT Press and McGraw-Hill (1986).
9. Hu, M.K., "Visual Pattern Recognition by Moment Invariants," IRE Trans. Inf. Theory, IT-8: 179-187 (1962).
10. Kim, W.S., Ellis, S., Tyler, M., Hannaford, B., and Stark, L., "Quantitatively Evaluation of Perspective and Stereoscopic Displays in Three-Axis Manual Tracking Tasks," IEEE System, Man and Cybernetics 16: 61-72 (1987).
11. Kim, W.S., Tendick, F., and Stark, L., "Visual Enhancement in Pick-and-Place Tasks: Human Operators Controlling a Simulated Cylindrical Manipulator", IEEE J. Rob. and Auto RA-3: 418-425 (1987).
12. Marr, D., Vision Freeman, San Francisco (1982).
13. Nguyen, A.H., Ngo, H., and Stark L., "Robotic Model Control of Image Processing", Proc. IEEE Intl. Conf. Sys., Man & Cyb., Beijing (1988)
14. Pratt, W. K. Digital Image Processing, Wiley (1978). <Y7
15. Reutz, P. and Brodersen, R., "An Image Recognition System Using Algorithmically Dedicated Integrated Circuits," Machine Vision Applic 1: 3-22 (1988).
16. Rosenfeld, A. and Kak, A. C., Digital Picture Processing, vols. 1 and 2, Academic (1982).
17. Sobel, I., "On Calibrating Computer Controlled Cameras for Perceiving 3-D Scenes," Art. Intel. 5: 185-198 (1974).
18. Stark, L., Kim, W., Tendick, F., et al., "Telerobotics: Display, Control, and Communication Problems," IEEE J Robotics Automation, RA-3: 67-75 (1987).
19. Noton, D., Stark, L. "Eye Movements and Visual Perception," Scientific American 224: 34-43 (1969).
20. Stark, L, Mills, B., Nguyen, A., and Ngo, H, "Instrumentation and Robotic Image Processing Using Top-Down Model Control", Robotics and Manufacturing, Jamshidi et al, eds., ASME, NY, 675-682 (1988).